International Journal of Safety Science
Vol. 01, No. 01, (2017), pp. 12-19
DOI:10.24900/01011219.2017.0301

Boolean Algebra Application in Simplifying Fault Tree Analysis

Yang Y* and Jung I
U.S. NRC
yaguang.yang@nrc.gov, ian.jung@nrc.gov

Abstract

This paper discusses Boolean algebra applications in fault tree analysis. Fault tree analysis has been extensively used in nuclear power plant safety analysis, such as analysis of system hazard, determination of critical characteristics in commercial grade dedication, and estimation of industrial system reliability. In general, the logic relations presented in fault tree models can be equivalently represented in Boolean algebra formulas. The Boolean algebra representation has several advantages over the original fault tree representation. The most significant one is that the Boolean representation can easily be simplified to get a so-called minimum cut representation. From there, fault tree analysis can be applied to several applications mentioned above. In this paper, we use some simple examples to demonstrate how to use Boolean algebra as a tool to simplify the fault tree model to get an expression of minimum cut. We then point out possible applications of this technique, such as common mode failure using identical digital system/components and remedy by diversity design, hazard analysis, and critical characteristics determination.

Keywords: *Boolean algebra, fault tree analysis, minimum cut representation*

1. Introduction

Fault tree analysis is an important branch in reliability and risk analysis theory. It has been investigated extensively in literatures (see for example and references therein) (Scott and Smalley 2003, Lee et al 1985 and Aven 1992). Fault tree analysis has many applications in nuclear industry (Ruijters and Stoelinga 2015 and Peplow et al 2004) and it is believed to be the most efficient way of handling the large logical models that are necessary for a nuclear power plant [Clause 4.154, McCormick 1981].

Because of the large logical models that are necessary to describe a nuclear power plant, there is a clear need to simplify the large logical models to get a logical expression which is easy to understand. In reference (IAEA 2001), a Boolean algebra method is proposed to achieve this goal. However, the main purpose of reference (IAEA 2001) is to derive a systematic method to estimate the reliability for digital instrumentation and control systems, which involves the estimation of both hardware reliability (Yang and Sydnor 2012) and software reliability (Bickel 2008).

In this paper, we will focus the details on how to simplify the large logical models using Boolean algebra method. We will demonstrate the efficiency of the this method in many applications, such as single failure event identification, Common Cause Failure (CCF) impact on system safety, hazard analysis, failure rate estimation, etc.

The remainder of the paper is organized as follows: Section 2 gives the detailed description of Boolean algebra method in fault tree model simplification. Section 3 provides various possible applications of the proposed logical simplification method. Conclusions are summarized in Section 4.

2. Fault Tree Model Simplification

First, we briefly review the basics about Boolean algebra.

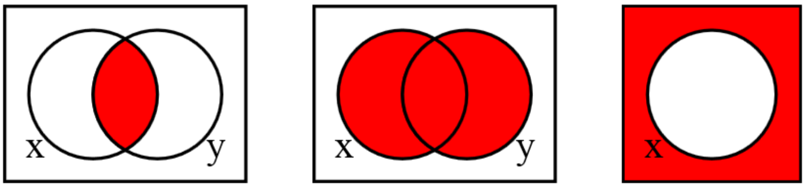
2.1. Boolean Algebra

Boolean algebra was named after George Boole who invented the algebra in his book (Yang 2009). The main operations of Boolean algebra are the conjunction *and* denoted as ∧, the disjunction *or* denoted as ∨, and the negation *not* denoted as ¬. A Boolean variable x can take only two values: 1 (or true) and 0 (or false). The values of $x\wedge y$, $x\vee y$, and $\neg x$ can be expressed by tabulating their values with truth tables as follows.

Table I. Truth Table

x	y	$x\wedge y$	$x\vee y$	x	$\neg x$
0	0	0	0	0	1
1	0	0	1	1	0
0	1	0	1		
1	1	1	1		

Using Venn diagram, we can express the above operations as in Fig. 1.



x∧y x∨y ¬x

Figure 1. Venn for Boolean algebra operations

Boolean algebra satisfies many of the same laws as ordinary algebra when one matches up ∨ with addition and ∧ with multiplication. In particular the following laws are common to both kinds of algebra:

(Associativity of ∨) $x\vee(y\vee z) = (x\vee y)\vee z$ (1)

(Associativity of ∧) $x\wedge(y\wedge z) = (x\wedge y)\wedge z$ (2)

(Commutativity of ∨) $x\vee y = y\vee x$ (3)

(Commutativity of ∧) $x\wedge y = y\wedge x$ (4)

(Distributivity of ∧ over ∨) $x\wedge(y\vee z) = (x\wedge y)\vee(x\wedge z)$ (5)

(Identity for ∨) $x\vee 0 = x$ (6)

(Identity for ∧) $x\wedge 1 = x$ (7)

(Annihilator for ∧) $x\wedge 0 = 0$ (8)

(Annihilator for ∨) $x\vee 1 = 1$ (9)

(Idempotence of ∨) $x\vee x = x$ (10)

(Idempotence of ∧) $x\wedge x = x$ (11)

(Absorption 1) $x\wedge(x\vee y) = x$ (12)

(Absorption 2) $x\vee(x\wedge y) = x$ (13)

However, distributivity of ∨ over ∧ is different from the ordinary algebra and is given as follows:

(Distributivity of ∨ over ∧) $x\vee(y\wedge z) = (x\vee y)\wedge(x\vee z)$ (14)

Besides, there are one double negation and two complement operation laws:

(Double negation) $\neg\neg x = x$ (15)

(Complementation 1) $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$ (16)

(Complementation 2) $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$ (17)

Finally, there are two De Morgan's laws:

(De Morgan 1) $(\neg x) \wedge (\neg y) = \neg (x \vee y)$ (18)

(De Morgan 2) $(\neg x) \vee (\neg y) = \neg (x \wedge y)$ (19)

2.2. Fault Tree Model Simplification Using Boolean Algebra

It will be easy to illustrate the method by using a simple example. Let consider an artificial digital I&C system as discussed in reference (IAEA 2001) as given in Fig. 2, which has three identical redundant smart sensors which have both hardware and software.

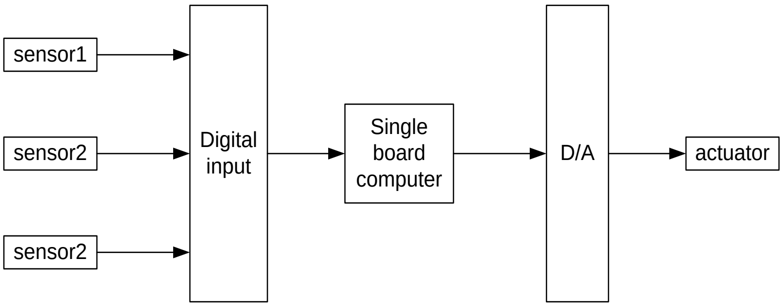


Figure 2. An artificial DI&C system

The measurements from the three sensors are sent to an A/D converter, the signal is processed in a single-board computer, and the control command is then sent to a D/A converter and then to an actuator. All components have two different failures, i.e., aging-related hardware failures and physical-damage related failures, except for the single-board computer and the smart sensors, which have two failure modes, i.e., aging related hardware failure and software failure. We also assume that the A/D always receives signals (correct or incorrect) from the three sensors while the signals are useful only if two of the sensors provide correct measurement.

The fault tree corresponding the digital I&C system can then be created as shown in Fig. 3, where, A, C, E, G, K, and M are aging-related failures; H, L, and N are physical-damage-related failures, I denotes computer hardware failures, and B, D, and F are software failures due to common-cause failure events X in smart sensors. J denotes software failures in the computer.

The failure tree model in Fig. 3 can be written in Boolean algebraic expressions as follows:

$$S = M \vee N \vee DA \tag{20}$$

$$DA = K \vee L \vee P \tag{21}$$

$$P = I \vee J \vee AD \tag{22}$$

$$AD = H \vee G \vee (S1 \wedge S2) \vee (S2 \wedge S3) \vee (S1 \wedge S3) \vee (S1 \wedge S2 \wedge S3) \tag{23}$$

$$S1 = A \vee B = A \vee X \tag{24}$$

$$S2 = C \vee D = C \vee X \tag{25}$$

$$S3 = E \vee F = E \vee X \tag{26}$$

Although Boolean algebra equations (20-26) provides a complete description of failure logic, this description is not the most convenient form in risk analysis. Using Boolean algebra formulas (1-17), we can reduce the Boolean algebra equations (20-26) into equivalent minimal cut set description which define all the "failure modes" of the DI&C failure events. First, from (24) and (25), we have:

$$\begin{aligned}
& S1 \wedge S2 \\
& = (A \vee X) \wedge (C \vee X) \\
& = [(A \vee X) \wedge C)] \vee [(A \vee X) \wedge X)] \\
& = [(A \vee X) \wedge C)] \vee [X \wedge (X \vee A)] \\
& = [(A \vee X) \wedge C)] \vee X \qquad (27) \\
& = X \vee [(A \vee X) \wedge C)] \\
& = [X \vee (A \vee X)] \wedge [X \vee C] \\
& = [X \vee A] \wedge [X \vee C] \\
& = X \vee (A \wedge C)
\end{aligned}$$

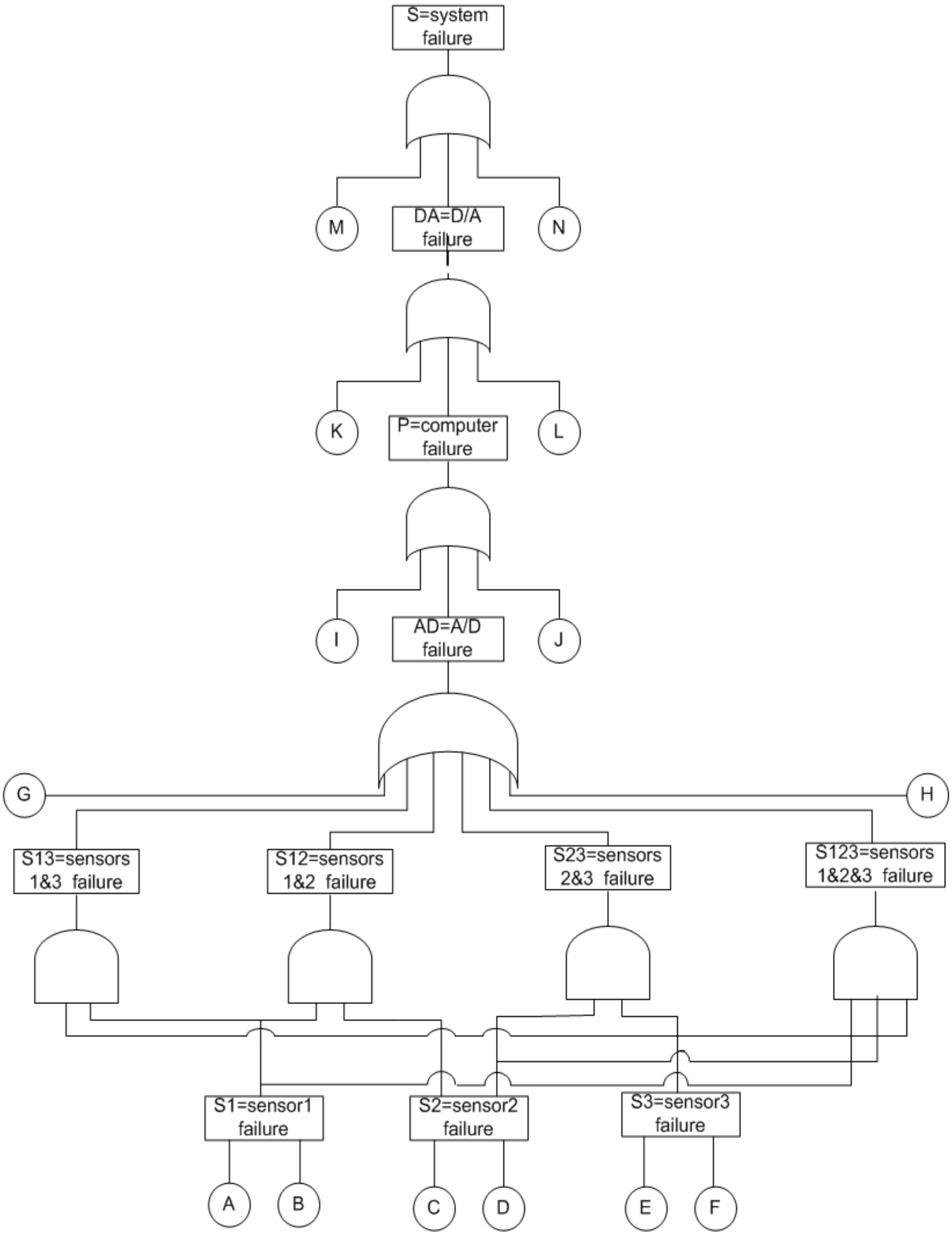


Figure 3. Fault tree of the DI&C system

Similarly, we can obtain

$$S1 \wedge S3 = X \vee (A \wedge E) \tag{28}$$

$$S2 \wedge S3 = X \vee (C \wedge E) \tag{29}$$

$$(S1 \wedge S2 \wedge S3) = X \vee (A \wedge C \wedge E) \tag{30}$$

This gives

$$\begin{aligned} &(S1 \wedge S2) \vee (S1 \wedge S3) \vee (S2 \wedge S3) \vee (S1 \wedge S2 \wedge S3) \\ &= [X \vee (A \wedge C)] \vee [X \vee (A \wedge E)] \vee [X \vee (C \wedge E)] \vee [X \vee (A \wedge C \wedge E)] \\ &= X \vee (A \wedge C) \vee (A \wedge E) \vee (C \wedge E) \vee (A \wedge C \wedge E) \end{aligned} \tag{31}$$

Using (20)-(26) and (31), we obtain the complete failure logic given as follows:

$$\begin{aligned} S &= M \vee N \vee DA = N \vee M \vee L \vee K \vee P \\ &= N \vee M \vee L \vee K \vee J \vee I \vee AD \\ &= N \vee M \vee L \vee K \vee J \vee I \vee G \vee H \vee (S1 \wedge S2) \\ &\vee (S2 \wedge S3) \vee (S1 \wedge S3) \vee (S1 \wedge S2 \wedge S3) \\ &= N \vee M \vee L \vee K \vee J \vee I \vee G \vee H \vee X \\ &\vee (A \wedge C) \vee (C \wedge E) \vee (A \wedge E) \vee (A \wedge C \wedge E) \end{aligned} \tag{32}$$

This simplification process looks tedious, but there are many software packages which can be used to handle the operations (Aven 1992).

3. Applications

The Boolean expression (32) is logically much clear and easier to be used in risk analysis relation applications than the fault tree expression in Fig. 3. From (32), we can see all possible failure modes are (1) single failure events, including N, M, L, K, J, I, G, H, and X, (2) double failure events, including $(A \wedge C)$, $(A \wedge E)$, and $(C \wedge E)$, and (3) triple failure event $(A \wedge C \wedge E)$. This logical failure modes were used in probabilistic risk analysis (PRA) in [7]. We will discuss several other applications in this section. For the sake of simplicity in our discussion/analysis, we assume *in the rest of section* that all failures, A, B, C, D, E, F, X, G, H, I, J, K, L, M, and N, have the same failure probability 10^{-4} per year and the probabilities are independent and identically distributed.

3.1. Single Failure Event

In nuclear industry, one of the very important safety system design criteria is Single Failure Criteria (Boole 1854). It basically says that the safety systems shall perform all safety functions required for a design basis event in the presence of any single detectable failure event. In this paper, the single failure event is defined differently. In the example discussed above, equation (32) indicates N, M, L, K, J, I, G, H, and X are single failure events in which any single event will disable the system to perform its function. We refer these events as to single failure events. But due to the redundancy and voting design for the smart sensors, the system will function correctly unless at least two of the three sensor hardware fail at the same time ($(A \wedge C)$ or $(A \wedge E)$ or $(C \wedge E)$ or $(A \wedge C \wedge E)$). We refer $(A \wedge C)$, $(C \wedge E)$, and $(C \wedge E)$ as to double failure events, and $(A \wedge C \wedge E)$ as to triple failure events, and so on.

3.2. Hazard and Risk Analysis

Although all failure events in the example of previous section, N, M, L, K, J, I, G, H, X, $(A \wedge C)$, $(A \wedge E)$, $(C \wedge E)$, and $(A \wedge C \wedge E)$, will cause the system to fail to perform its function, and should be considered in hazard analysis, the risk of these events are different. For all single failure events, N, M, L, K, J, I, G, H, X, as we have assumed, their failure probabilities are 10^{-4} per year. But the failure probabilities for the double failure events are $p(A \wedge C) = p(A)p(C) = p(A \wedge E) = p(C \wedge E) = 10^{-8}$ per year; the failure probability of triple failure events $p(A \wedge C \wedge E) = 10^{-12}$ per year. Therefore, redundancy and voting design for smart sensor hardware failure reduces the risk significantly (10^{-4} vs 10^{-8}). That is the main reason that single failure events are most risk events and should be considered seriously.

3.3. CCF and Diversity Consideration

Another serious safety concern in nuclear industry is Common Cause Failure (CCF) scenario in safety system. In the example discussed in the previous section, if all three smart sensors use the same hardware and identical software, then a bug in the software can trigger a CCF event X. Based on the analysis in the previous section, this event will prevent the system from performing it function, which has the failure probability of 10^{-4} per year. However, if the smart sensors use three different software packages to conduct the same function, the system can be modeled as three different failure events, B, D, and F. With this design diversity consideration, we can show that CCF will not be a concern in this scenario. Indeed, equation (27) in this scenario becomes

$$\begin{aligned} &S1 \wedge S2 \\ &= (A \vee B) \wedge (C \vee D) \\ &= [(A \vee B) \wedge C[\vee[(A \vee B) \wedge D] \\ &= (A \wedge C) \vee (C \wedge B) \vee (A \wedge D) \vee (B \wedge D) \end{aligned} \tag{33}$$

Similarly, we can obtain

$$S1 \wedge S3 = (A \wedge E) \vee (A \wedge F) \vee (E \wedge B) \vee (B \wedge F) \tag{34}$$

$$S2 \wedge S3 = (C \wedge E) \vee (C \wedge F) \vee (E \wedge D) \vee (D \wedge F) \tag{35}$$

$$\begin{aligned} S1 \wedge S2 \wedge S3 = {} & (A \wedge C \wedge E) \vee (A \wedge C \wedge F) \vee (A \wedge E \wedge D) \vee (A \wedge D \wedge F) \\ & (B \wedge C \wedge E) \vee (B \wedge C \wedge F) \vee (B \wedge E \wedge D) \vee (B \wedge D \wedge F) \end{aligned} \tag{36}$$

Note that we omitted some higher order terms in (36) which have very little impact in risk analysis if any. The complete failure logic in this scenario becomes

$$\begin{aligned} S = {} & N \vee M \vee L \vee K \vee J \vee I \vee G \vee H \vee (A \wedge C) \vee (C \wedge E) \vee (A \wedge E) \\ & \vee (A \wedge D) \vee (A \wedge F) \vee (C \wedge B) \vee (C \wedge F) \vee (E \wedge B) \vee (E \wedge D) \\ & \vee (B \wedge D) \vee (B \wedge F) \vee (D \wedge F) \\ & \vee (A \wedge C \wedge E) \vee (A \wedge C \wedge F) \vee (A \wedge E \wedge D) \vee (A \wedge D \wedge F) \\ & \quad \vee (B \wedge C \wedge E) \vee (B \wedge C \wedge F) \vee (B \wedge E \wedge D) \vee (B \wedge D \wedge F) \end{aligned} \tag{37}$$

Therefore, the smart sensor software failure is no longer a single failure event in this scenario. Although, there are a few more double and triple failure events, these double and triple failure probabilities are much lower than the single failure probabilities in the

CCF failure event. This shows how much the diversity design consideration reduces the risk of the smart sensor failure (10^{-4} vs 10^{-8}).

3.4. Commercial Grade Dedication and Critical Characteristics

In nuclear industry, all components and systems used in safety system should be designed and manufactured by following the process described in Code of Federal Regulations, 10 CFR Appendix B (IEEE standard 603 2009), which has very strict requirements on design and manufacture documentations. However, these requirements are very difficult to be satisfied for components and systems available in today's market where components and systems are not designed and manufactured following the process of Appendix B. An alternative process with less documentation requirements, called commercial grade dedication, is therefore introduced in (Part 21, IEEE standard 603 2009). For a commercial grade item (meaning a structure, system, or component, or part thereof that affects its safety function, that was not designed and manufactured under a quality assurance program complying with appendix B to part 50 IEEE standard 603 2009), an important concept in this process is critical characteristics (Part 21, IEEE standard 603 2009] which is defined as: "critical characteristics are those important design, material, and performance characteristics of a commercial grade item that, once verified, will provide reasonable assurance that the item will perform its intended safety function." Therefore, we may consider all single failure events as part of critical characteristics. If the probability of these failure events is verified to be small enough, it will provide reasonable assurance that the commercial grade item will perform its intended safety function because the failure probabilities for higher order failure events are much small and therefore can be ignored. This idea is similar to the EPRI's proposed method of using failure modes and effects analysis method to determine the critical characteristics [Section 1.5.3, Code of Federal Regulations 2014], but we provides an easy to implement and mathematically rigorous method rather than a general method.

4. Conclusions

In this paper, we proposed a method to use Boolean algebra to simplify the fault tree model to obtain all failure modes of a structure or a system or a component. The result obtained from this method provides a clear description of all failure modes of the structure, or the system or the component. It is easy to see that single failure events are the most risk events and need to have special consideration in the safety analysis. This method can be applied to several problems related to the safety and risk analysis, such as, probabilistic risk analysis, hazard and risk analysis, CCF analysis, and commercial grade dedication in nuclear industry.

References

[1] C. D. Scott and R. E. Smalley, "Diagnostic Ultrasound: Principles and Instruments", Journal of Nanosci. Nanotechnology., vol. 3, no. 2, **(2003)**, pp. 75-80.

[2] W.S. Lee, D.L. Grosh, F.A. Tillman, and C.H. Lie, "Fault tree analysis, methods, and applications—a review," IEEE Transactions on reliability, vol. 34, **(1985)**, pp. 194-203.

[3] T. Aven, "Reliability and risk analysis", Elsevier Applied Science, London and New York **(1992).**

[4] E. Ruijters and M. Stoelinga, "Fault tree analysis: A survey of the state-of-the-art in modeling, analysis and tools," Computer Science Review, vol. 15-16, **(2015)**, pp. 29-62.

[5] D. E. Peplow, C. D. Sulfredge, R. L. Sanders, R. H. Morris, and T. A. Hann, "Calculating Nuclear Power Plant Vulnerability Using Integrated Geometry and Event/Fault Tree Models," Nucl. Sci. Eng. vol. 146(1), **(2004),** 71–87.

[6] N. J. McCormick, "Reliability and risk analysis: methods and nuclear power applications", Academic Press, San Diego, CA (**1981**).

[7] IAEA, "Safety Assessment and verification for nuclear power plants", IAEA Safety Standards Series No. NS-G-1.2, Vienna **(2001).**

[8] Y. Yang and R. Sydnor, "Reliability estimation for a digital instrumentation and control system," Nuclear Engineering Technology, vol. 44, **(2012),** pp. 405-414.

[9] J. H. Bickel, "Risk Implications of Digital Reactor Protection System Operating Experience," Reliability Engineering & System Safety, vol. 93, **(2008)**, pp. 107-124.

[10] Y. Yang, "A flow network model for software reliability assessment," Proceeding of 6th American nuclear society international topical meeting on nuclear plant instrumentation, control, and human-machine interface technologies, Knoxville, **(2009)**, April 5-9.

[11] G. Boole, An investigation of the laws of thought, Macmillan and Co., London **(1854)**.

[12] IEEE standard 603, Criteria for safety systems for nuclear power generating stations, IEEE Standard 603, New York **(2009)**.

[13] Code of Federal Regulations, "10 CFR", Office of the Federal Register, **(2014).**

[14] EPRI, Plant Engineering: Guideline for the Acceptance of Commercial-Grade Items in Nuclear Safety-Related Applications, Revision 1 of NP-5652 and TR-102260, **(2014).**